网络应用开发课程设计

网页游戏（第三组二阶段报告）

**组员信息：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 姓名 | 学号 | 成员贡献 | 课程设计成绩 |
| 1 | 冯许晔 | 2021212162 | 代码实现、报告撰写 |  |
| 2 | 冯祎盈 | 2021212191 | 代码实现、报告撰写 |  |
| 3 | 张密 | 2021212195 | 代码实现、报告撰写 |  |
| 4 |  |  |  |  |
| 5 |  |  |  |  |

Github代码仓库URL

# **引言**

参考贪吃蛇游戏，设计一款迪士尼角色收集游戏。可供单人游戏也可供二或三人进行小游戏比赛，进入页面后玩家先选择代表自己的角色（Mickey/Minnie/Donald）再点击开始游戏，“米奇”作为初始角色移动，玩家可以通过键盘“上、下、左、右”控制角色移动，游戏画布上会随机生成“许愿星”，“米奇”需要通过收集“许愿星”增加角色，且玩家的得分会增加，若角色移动时角色碰到游戏界面则游戏结束，若得分达到25则游戏结束该玩家胜利。

# 功能与设计

## 功能介绍

模块移动：玩家可以通过键盘或其他控制方式控制模块的移动方向，例如上、下、左、右。

“许愿星”生成：游戏界面上会随机生成“许愿星”，在上一个“许愿星”被收集后随机生成下一个。

碰撞检测：在每次角色移动，需要检测角色是否与“许愿星”或者游戏边界发生碰撞。

得分和计分板：游戏界面通常会显示玩家的得分，即收集的“许愿星”的数量，得分可以作为游戏的评价标准，玩家可以尽量收集更多的“许愿星”，获得更高的得分。计分板记录每位玩家多次游戏中的最高分。

游戏结束和重新开始：当角色碰撞到游戏界面的边界时，游戏结束，玩家可以选择重新开始游戏，重新生成角色，并且清零得分，或者退出游戏。

## 系统设计

### 2.2.1系统运行的逻辑和关键设计

初始化游戏：

创建游戏画布和绘图上下文

设置单元格的大小

定义角色的初始位置和速度

定义“许愿星”的初始位置

定义得分初始值为 0

监听用户输入：

监听键盘事件，根据按键改变角色的移动方向

更新游戏状态：

在每个游戏循环中，根据当前的移动方向，将角色移动到新的位置

判断是否收集“许愿星”：

如果角色的位置与“许愿星”位置相同，表示收集

生成新的“许愿星”位置

得分增加，角色增加

绘制游戏画面：

清空画布

绘制角色，绘制“许愿星”

绘制得分

游戏主循环：

在每个游戏循环中，依次执行以下步骤：

更新游戏状态

绘制游戏画面

请求下一帧动画

启动游戏：

监听开始按钮的点击事件。

点击开始按钮后，调用游戏主循环函数，开始游戏

### 2.2.2系统架构

用户界面：显示游戏画面，包括角色、“许愿星”、分数。

游戏逻辑：控制游戏规则，如角色的移动、“许愿星”的生成和分数的计算。

输入处理：处理玩家输入，转换为角色模块的移动指令。

数据存储：存储玩家分数

### 2.2.3逻辑流程

初始化游戏→游戏循环（包括处理输入、更新游戏状态）→游戏结束

### 2.2.4通信协议

HTTP/HTTPS： 在初始加载网页时，浏览器通过HTTP或HTTPS请求从服务器获取HTML、CSS和JavaScript等资源。

# **功能实现**

许愿星生成：随机生成食物，位置不能与模块重叠。

碰撞检测：角色移动时，需要检测角色是否与“许愿星”及游戏边界发生碰撞，当角色碰到“许愿星”，“许愿星”会被收集，并在游戏画布上生成新的“许愿星”；若角色碰到游戏边界，则游戏结束。

最高得分记录：再不重新加载网页的情况下，同一玩家多次游戏取最高分记录。

# **功能展示与评测**

（进入游戏界面）



（游戏界面）

（整体呈现视频）



# **总结**

二阶段实现一阶段的功能设计，丰富了画面，增加了记录最高分的功能。搭建该网页游戏巩固了对前端开发基础的理解，也对如何将代码转化为实际的用户体验有了更深的认识。通过实际操作，学会了如何将复杂问题分解为可管理的小部分，并逐步实现功能。